(1)

National Defence / Défense nationale

ADA223 219

AN ALGORITHM FOR AUTOMATED SURFACE MODELLING IN UNIGRAPHICS (U)

by

D. Hidson
Chemical Protection Section
Protective Sciences Division

DEFENCE RESEARCH ESTABLISHMENT OTTAWA

TECHNICAL NOTE 90-6

PCN
051LD11

March 1990
Ottawa

ABSTRACT

Remote laser scanning of complex 3-D geometry can generate data sets with large numbers of points. This paper describes how these data are read into a CAD/CAM database (Unigraphics). A program is written in Graphics Interactive Programming Language (a subset of Unigraphics) that edits the data files, generates points, spline curve and surface entities in Unigraphics and stores the created models and parts in the databases. Programs for data sampling and editing are also shown separately.

RÉSUMÉ

Le balayage à distance par laser de géométrie complexe à 3 dimensions peux générer des groupes de données contenant un grand nombre de points. Cette note technique décrit comment ces données sont transformées en une banque de données CAO/FAO (Unigraphics). Un programme fut écrit utilisant le Graphics Interactive Programming Language (une sous-routine de Unigraphics) qui permet d'éditer les fichiers, de génereer des points, des courbes non paramétriques et des profils de surface en Unigraphics et de stocker les modèles créés de même que lesdifférentes composantes dans des banques de données. Les programmes d'échantillonnage et d'édition des données sont aussi presentés séparément.

EXECUTIVE SUMMARY

This paper describes programs written in Graphics Interactive Programming Language (GRIP) to generate models in the Unigraphics CAD/CAM database of complex three dimensional geometry relating to human body shape.

Data from human faceform models gathered by remote laser sensing was transferred to the CAD/CAM system. The large data files (30,000 points) were edited down to more manageable sizes of approximately 3,000 points. Programs were written in GRIP (a) to sort points into regular arrays, (b) to generate complete sets of orthogonal spline curves spanning the sets of points, (c) to create sets of sculptured surface patches covering the data and (d) to display and store the resulting information.

The final surface renderings are shown as displays on a vector refresh terminal. GRIP source code is included as appendices.

The software allows for the generation of sculptured surfaces from any regular, point data set.

TABLE OF CONTENTS

1.0 INTRODUCTION

The Chemical Protection Section at the Defence Research Establishment Ottawa is involved in work dedicated to providing equipment to protect the Canadian Forces from the immediate effects of nuclear, chemical and biological (NBC) warfare. The work is devoted almost exclusively to the development of protective clothing such as boots, respirators, gloves, canisters etc. that can protect the soldier from the toxic agents and biological contaminants likely to be encountered in an NBC war environment.

In designing protective clothing, gloves, masks and other items of personal equipment, it is essential to acquire large quantities of anthropometric data that describe the sizes and shapes of the human form. From these data, various design criteria can be established that define the average or design dimensions of the part to be made. Computers can now be used effectively to model the data and render them in the form of images on a graphics terminal.

All anthropometric data in the past were gathered by means of tape measures and calipers and a large pool of human subjects. This process was very time consuming and expensive. Recent work sponsored by DREO (1) has indicated that several other methods of gathering these data offer much more promise for the future. Remote sensing by means of laser scanning or sonic digitization are two methods of interest. Both of these techniques (among others) produce large quantities of digital data in a form ready to be read into, and modelled by, a computer.

Further work on remote scanning techniques by laser has been carried out on a trial basis (2) and the data stored in digital form on magnetic tape. The laser scanning technique may produce up to a million data points in a few seconds of scanning. The time taken for the production of scan data for the human body is now of the order of one second - illustrating the fact that it is quite practical as a technique for gathering anthropometric data.

Data were gathered with this technique using four anthropometric models (faceforms) as targets. These faceform models were used as a design base for the C4 mask and they were made by the Anthropology Research Project, Yellow Springs, Ohio, USA (3). The laser scanning device was designed and manufactured by Hymarc Engineering of Ottawa, Canada, and used a helium-neon laser as the light source. A scanning resolution of 1.0 mm was used which, over the complete set of faceforms, produced about 30,000 data points per target (2). The data were presented in the form of digital data on a magnetic tape from a VMS operating system environment.

These data had to be converted for use on the Data General MV4000 computer at DREO which runs the Unigraphics applications software. After the data had been converted, that is, the word structures changed to be readable by the AOS/VS operating system, further editing of the data format was required. Quantities of data points had to be trimmed to the one or two thousand usable by the CAD design process. Then the data were reformatted and fed into the Unigraphics environment where GRIP (Graphics Interactive Programming Language) programming could further analyze them.

Software was written to sample and edit the format of the digital data provided by the laser scans, to sample the data yet again into the required subsets, and to construct automatically geometric entities (points, spline curves and sculptured surfaces) to model the data and generate the surfaces.

The algorithm developed was written in GRIP, a module of the Unigraphics software, so that it would be applicable to all Unigraphics CAD/CAM facilities regardless of the hardware platform on which the applications run.

2.0 DATA ACQUISITION AND EDITING

The data were gathered using a Hyscan 60 Camera manufactured by Hymarc Engineering of Ottawa. In this device, a pair of synchronized rotating mirrors both send and receive the laser beam. The beam is deflected by a mirror onto the object and the reflection picked up by a photodetector. The angle of the mirrors, the position of the light spot on the object and the coordinates of the detector are digitized and computer controlled. This entire assembly may be moved around the object under study. The camera assembly was mounted on a robot arm and two separate scans were taken so that the sides of the head could be fully defined and the problems of shadowing avoided. The scans were taken approximately 45 degrees on either side of the center line. Some problems associated with the accuracy of the landmarks on the faceforms were noted and this subject forms the basis of some further work.

Five files were presented for each scan of the faceform: **headn.img**, an integrated image of faceform #n; **headn.01**, a gridded image of faceform #n; **headndec.img**, a one-ninth undersampled file of faceform #n(*.01); **headn0.bsp**, a gridded image of faceform #n - offset in Y; and **headnsm.bsp**, a smoothed, gridded image of faceform #n(*0.bsp). The term 'gridded image' refers to the layout of data points in the image system. The points are recorded sequentially with regular spacing in the x- and y- dimensions. This allowed for easier mathematical manipulation of the data into arrays and their subsequent use in the surface generation operations.

The files used as source files were the **headndec.img** set, the one-ninth undersampled set, which still contained more than sixteen thousand points. Further, these data were delivered on tapes produced by the VMS operating system of a Digital Equipment Corporation VAX computer and had to be converted to be readable by the AOS/VS operating system on the Data General computer that was the platform for the Unigraphics CAD/CAM software.

This was done in the following way. The tapes were mounted on the Data General MV4000 computer and the files read into the memory using the COPY command. It was essential that the data were stored on the VMS tape also using the VMS COPY command. Files stored by a general backup command on the VMS system could not be read or copied in any way by the AOS/VS system.

Copying one VMS file to AOS/VS results in the production of three files; the first contains the file header information; the second contains the data or the body of the information contained in the original; and the third the file terminator. Only the second file need be kept; the others can be discarded.

Once the files were loaded into the memory of the MV4000, they could be edited. First, the routine CONVERT_1(4) was used to change the word structure of the data to match that of the MV4000 architecture. VAX/VMS files produced by the COPY command possess a four-byte ASCII header when the file is transferred from disc to tape. The Data General MV4000 expects to receive data-sensitive records (terminated by <NEW LINE>) and a two-byte binary header. Further, the file in blocks of 2048 bytes will have the unused portion of each block filled with carets. These must also be deleted in the process of making the files fully readable by all AOS/VS system functions. This the CONVERT_1 routine accomplishes as well. It will not perform such necessary changes as VIRTUAL to DIMENSION changes as may be required in program source files.

3.0 PROGRAM DEVELOPMENT

In the Unigraphics environment, the PUT and GET commands under the file maintenance module, FMEXEC, which is the executive file manager, were used to transfer the data between the AOS/VS operating system and the users Unigraphics files.

Again, there was almost too much data to handle, so another sampling of the dataset was performed using GRIP (Graphics Interactive Programming Language). This reduced the number of points to a more manageable quantity of less than four thousand.

The data are displayed in a grid in the x- and y- dimensions with the dependent variable being in the z-direction. The program NUSAMPL.GRS (Appendix A) extracts every n-th point it reads from the source file (governed by the variable MVAL) in the x-direction

and skips the same number of rows in the y-direction (governed by the variable NVAL). This produces a lower density grid of points and the density of points is uniform along both axes.

The output file, NUSAMPL.TXT, forms the data source for the subsequent analyses in POINTGEN.GRS and MULTI_PATCH.GRS. Files with the suffix ".TXT" contain ASCII information, usually data. Those ending ".GRS" contains GRIP source code for executable programs and those with the suffix ".PRT" are Unigraphics part files in the CAD database.

POINTGEN.GRS (see Appendix B) performs the function of mapping and displaying the points in a data file into a Unigraphics part. The logic path is shown in Figure 1. This program eventually forms part of MULTI_PATCH.GRS (see Appendix C) which generates points, spline curves and sculptured surfaces. As it stands alone, POINTGEN.GRS will quickly delineate the point data set allowing a preliminary perusal of the shape and the integrity of the data. Figure 2 shows two views of the point data NUSAMPL.TXT, resulting from the execution of the program POINTGEN.GRS. An isometric and a side view are shown.

MULTI_PATCH.GRS combines the abilities of both the above routines and generates the surfaced model. The variables are defined and the part (MULTI_PATCH.PRT) created in the database. The source data are contained in the file NUSAMPL.TXT and are brought into the main program by means of the FETCH command. The format of the READ statement contained in the program must match that of that source data file unless data sensitive records are kept. In this case, the delimiters (either a blank or a comma) must be used to separate the fields and they must be specified. The exact number of data points must be known as they are read into a two-dimensional array, PTA(I,J). The data array is then subdivided into PATCH(P,Q) subarrays which then form the basis for the spline curve construction.

One should note that any program in GRIP that attempts to create a part file during execution will produce errors if a part with that name already exists in the database. Further, if the program is executed in the foreground (i.e. it occupies the CPU and the terminal), there must be no part file in the active work space. This will generate file creation errors.

The size of the smaller arrays (PATCH(P,Q)) is arbitrary. Large surfaces can be constructed but future modification and machining are easier if only parts of the surface have to be changed. So there is a trade-off between the size of the sculptured surfaces created and the number of these surfaces. In the program the variables 'V' and 'W' define the patch surface size in terms of the number of points along each curve.

A second reason for defining PATCH(P,Q) is to subdivide the machining computation. Sculptured surfaces with complex geometry require relatively large amounts of computing time when the cutter paths are calculated. Errors can be corrected more easily and require less re-computation if the individual patches are of a more manageable size.

Finally, the spline curves are transformed using a translation matrix, MAT1, to display the spline curves over the surface display. The VIEWE/AUTO command resizes the display to fit the screen space. A schematic of the process is shown in Figure 3. The complete program is shown in Appendix C.

In Figure 4 the set of patch surfaces that were created is displayed in a tri-metric view. Figure 5 shows the complete set of spline curves that were constructed to define the surface: these are also shown in a tri-metric view. Figure 6 displays the gridded surface view of the whole structure. Note that the spot labelled by the arrow shows some surface inaccuracy due here to the high surface gradient at the edge of the model. Figure 7 shows two views of the spline curve surface structure and Figure 8 shows four views: a top, side, tri-metric and isometric.

Figure 9 displays a close-up view of the front facial area based on the spline display.

4.0 CONCLUSION

An algorithm was developed for sampling large data files containing point-to-point data from remote sensing techniques (NUSAMPL.GRS). MULTI_PATCH.GRS was designed to read these data and model automatically the large numbers of points, sort them into data arrays and construct spline curves in orthogonal sets. It then generates the surface patches (of arbitrary shape and size) to produce a surfaced model ready for modification or machining. In the work being carried on at DREO on shape analysis and averaging it can be used to generate shapes from the laser scan data. The faceform used in the example shown was the extra small version. For producing models by means of a Computer Numerical Controlled Milling machine, the machining process may also be rendered in GRIP.

5.0 REFERENCES

(1) Gallup B, Shewchenko N, et al, "A Final Report on the Quantification and Summarization of Human Body Shape", Biokinetics and Associates, Document R87-30, December 1987, Ottawa.

(2) Gallup B, Shewchenko N, "Laser Digitization of Anthropometric Faceforms", Biokinetics and Associates, Document R87-27, November, 1987, Ottawa.

(3) ARP, Inc, "The Development of an Integrated Anthropometric Sizing System for Respirator Design", DREO, Project D6369, Serial No. 8PW84-00052.

(4) Data General, "FORTRAN 77 Reference Manual", section G41, 093-000-162-03, 1985.

POINTGEN.GRS

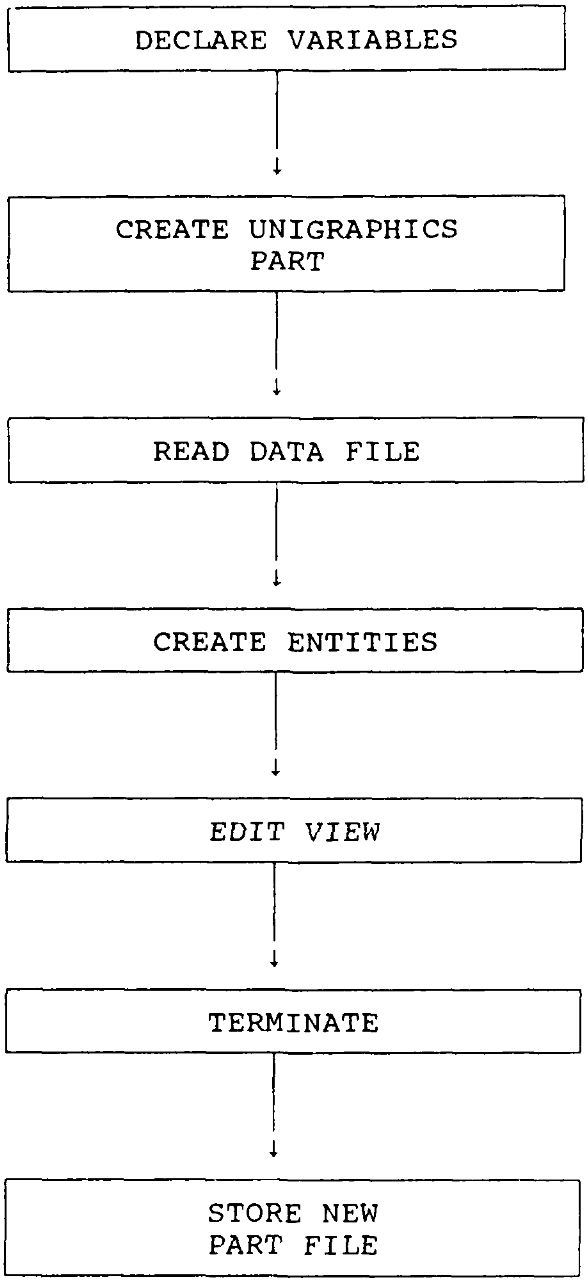


FIGURE 1: Flowchart of POINTGEN.GRS Program.

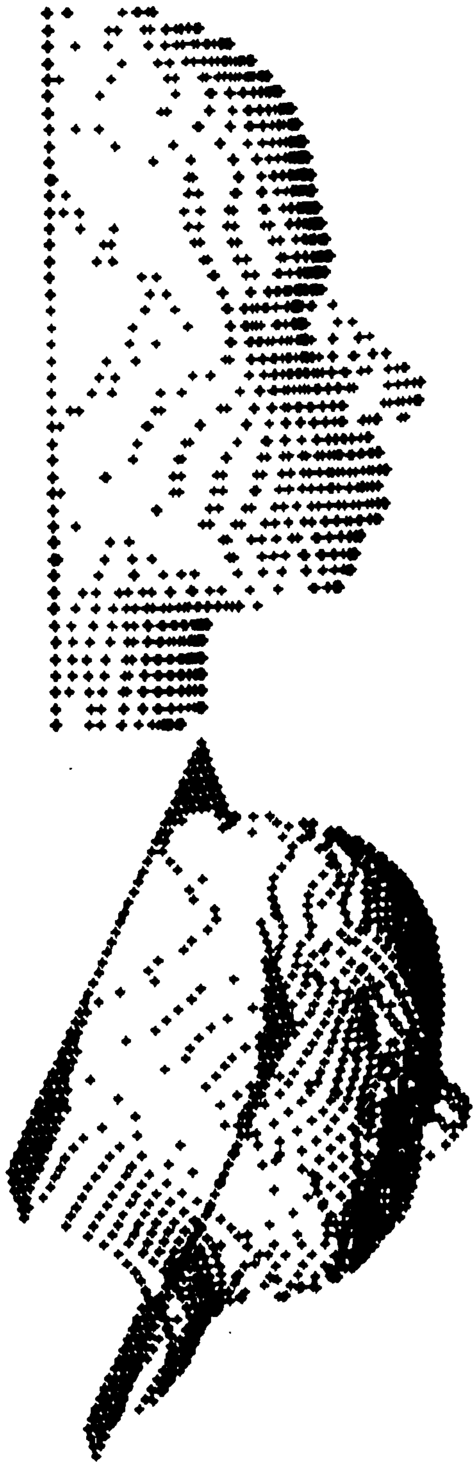
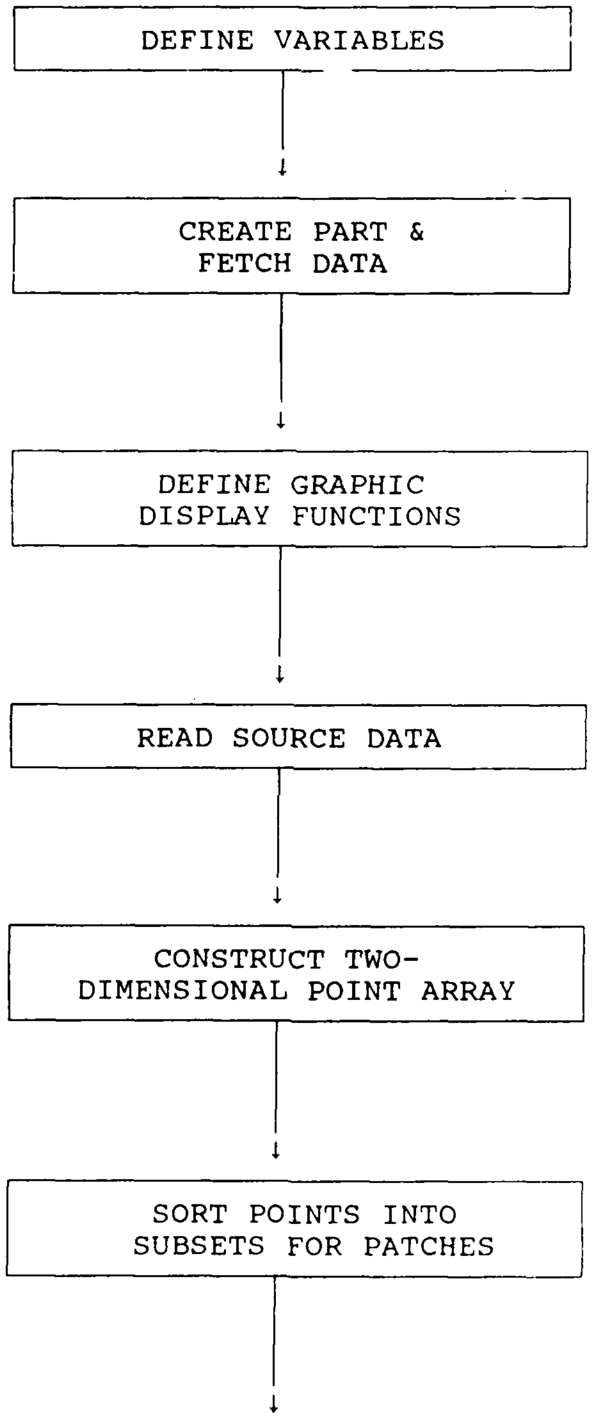


Figure 2: Two views of point display of NUSAMPL.TXT

MULTI_PATCH.GRS



(CONT'D)

MULTI_PATCH.GRS (cont'd)

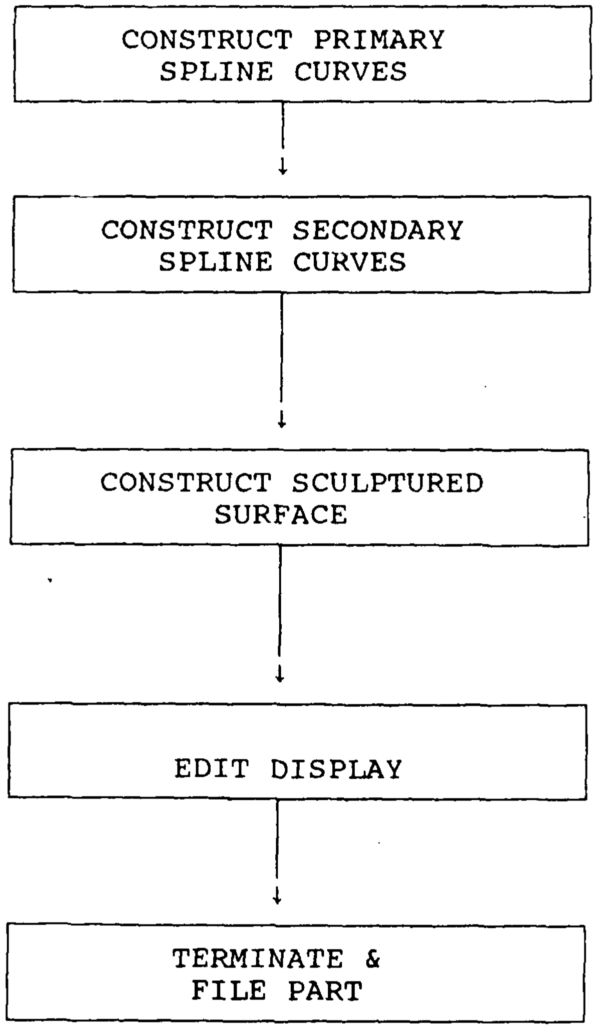


FIGURE 3: Flowchart of MULTI_PATCH.GRS Program

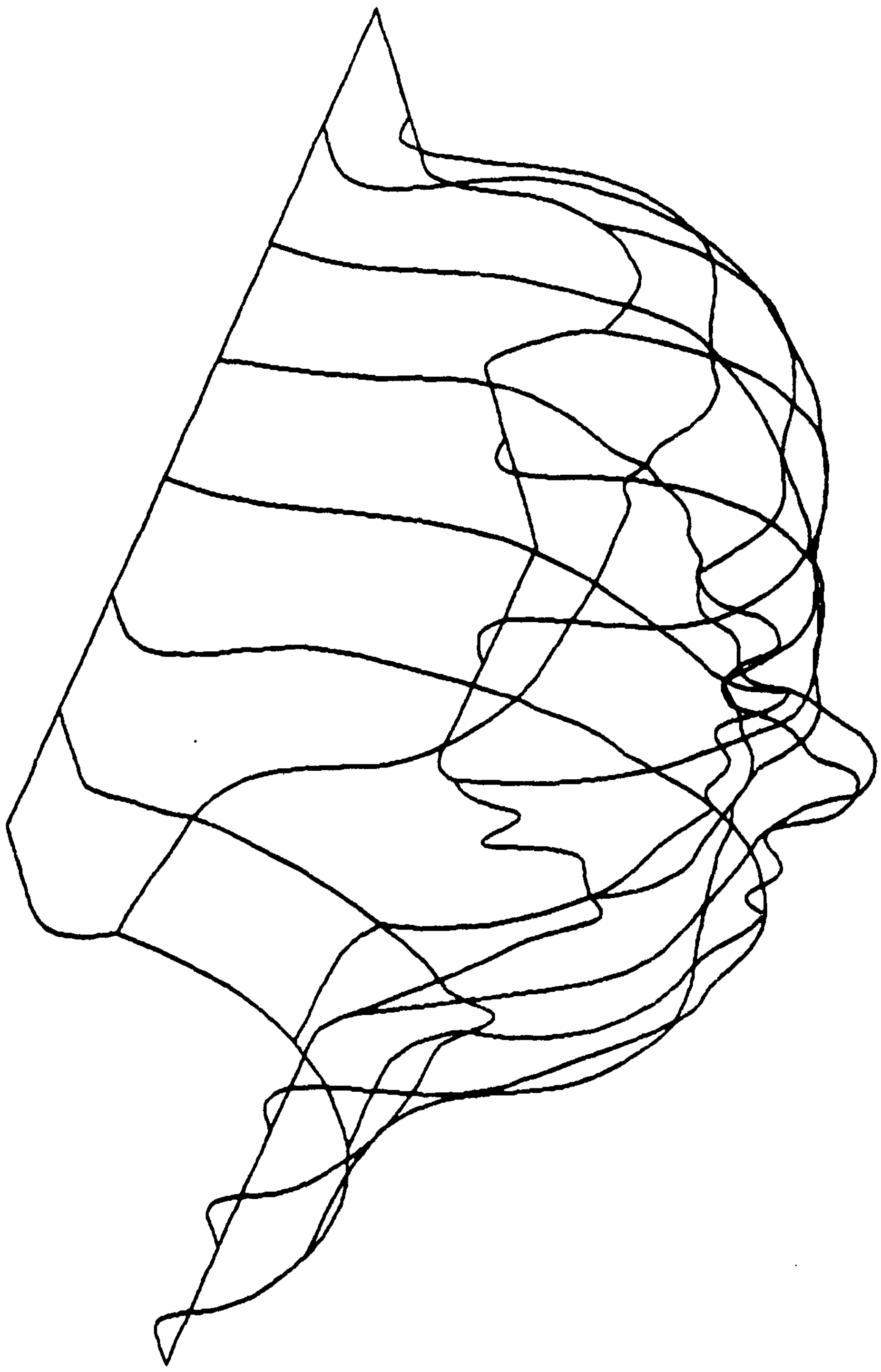


Figure 4: Tri-metric view of surfaces making up MULTI_PATCH.PRT

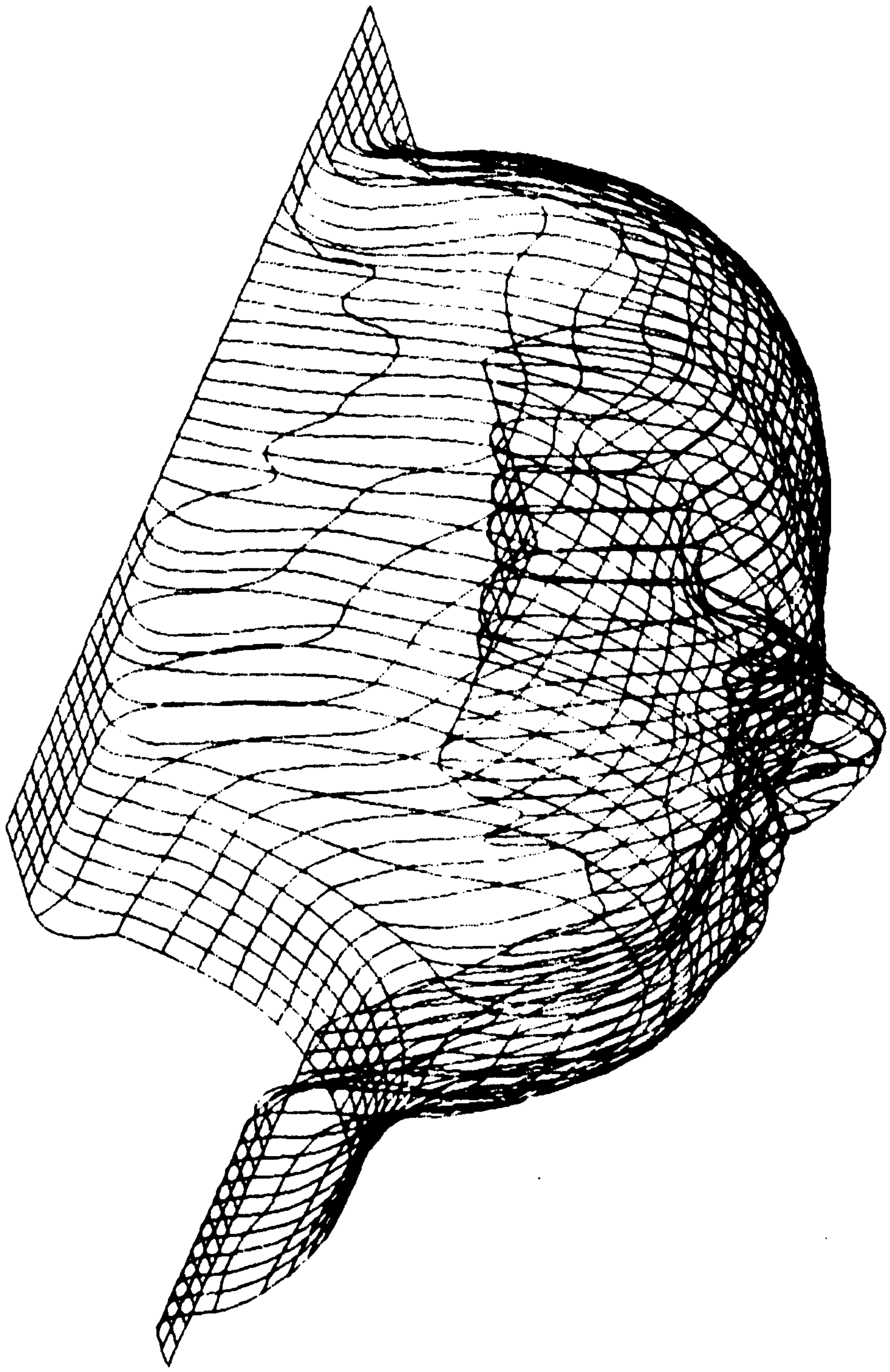


Figure 5: Tri-metric view of spline curves making up MULTI_PATCH.PRT

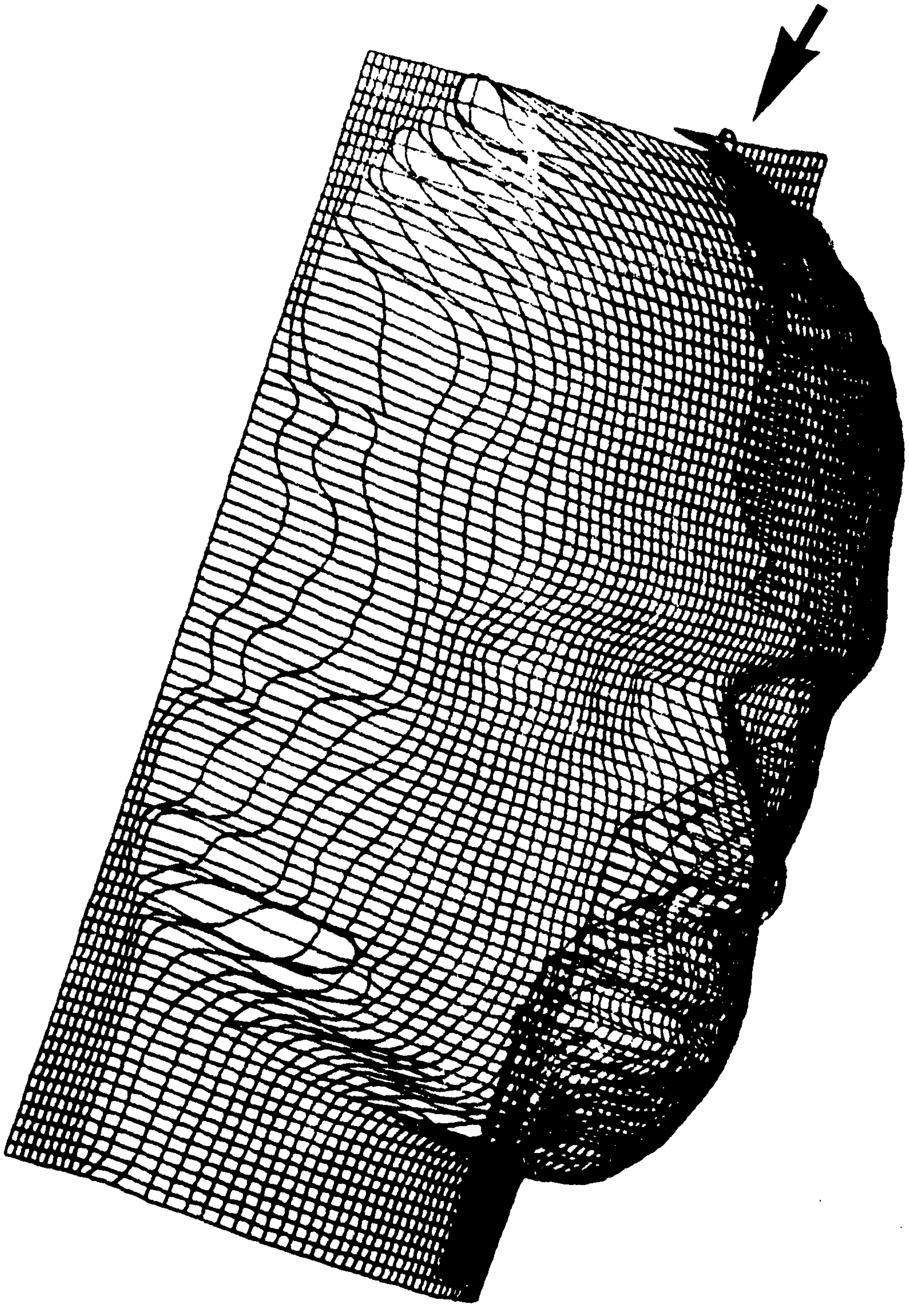


Figure 6: View of surfaced-model of MULTI_PATCH.PRT

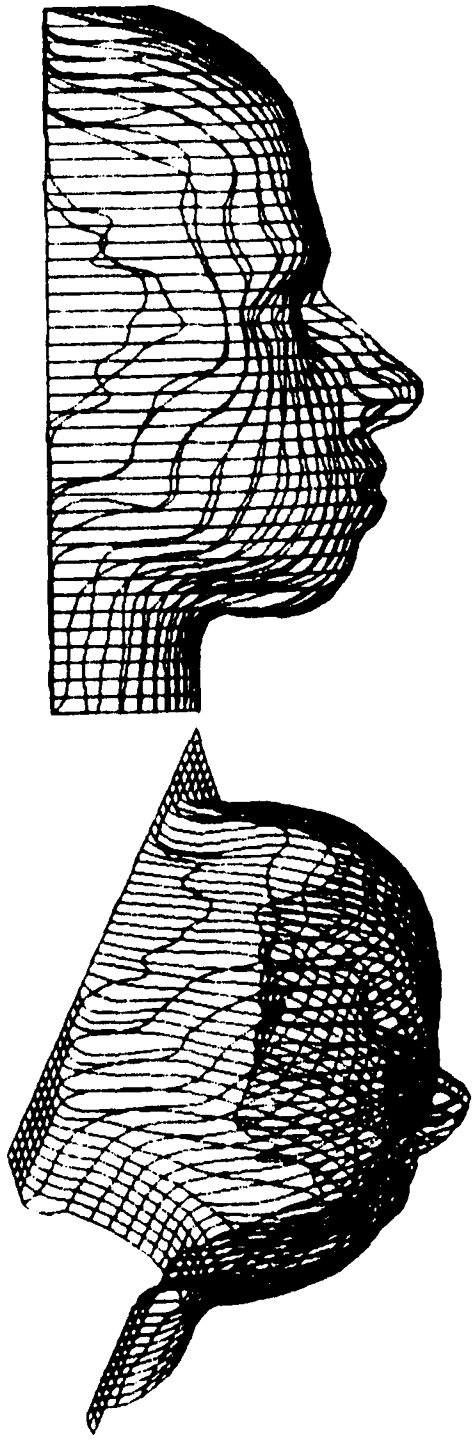


Figure 7: Side and isometric views of MULTI_PATCH.PRT

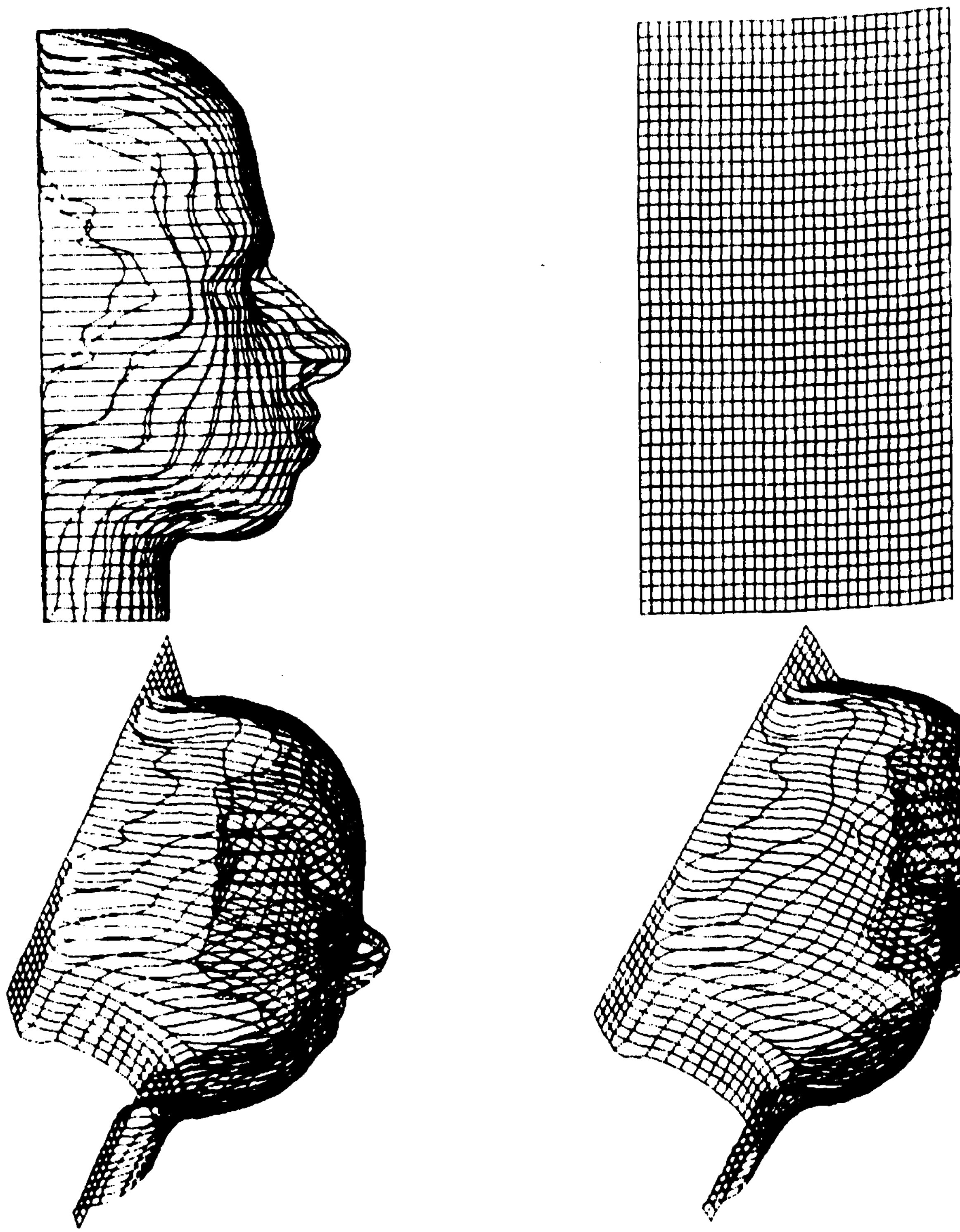


Figure 8: Four views of MULTI_PATCH.PRT

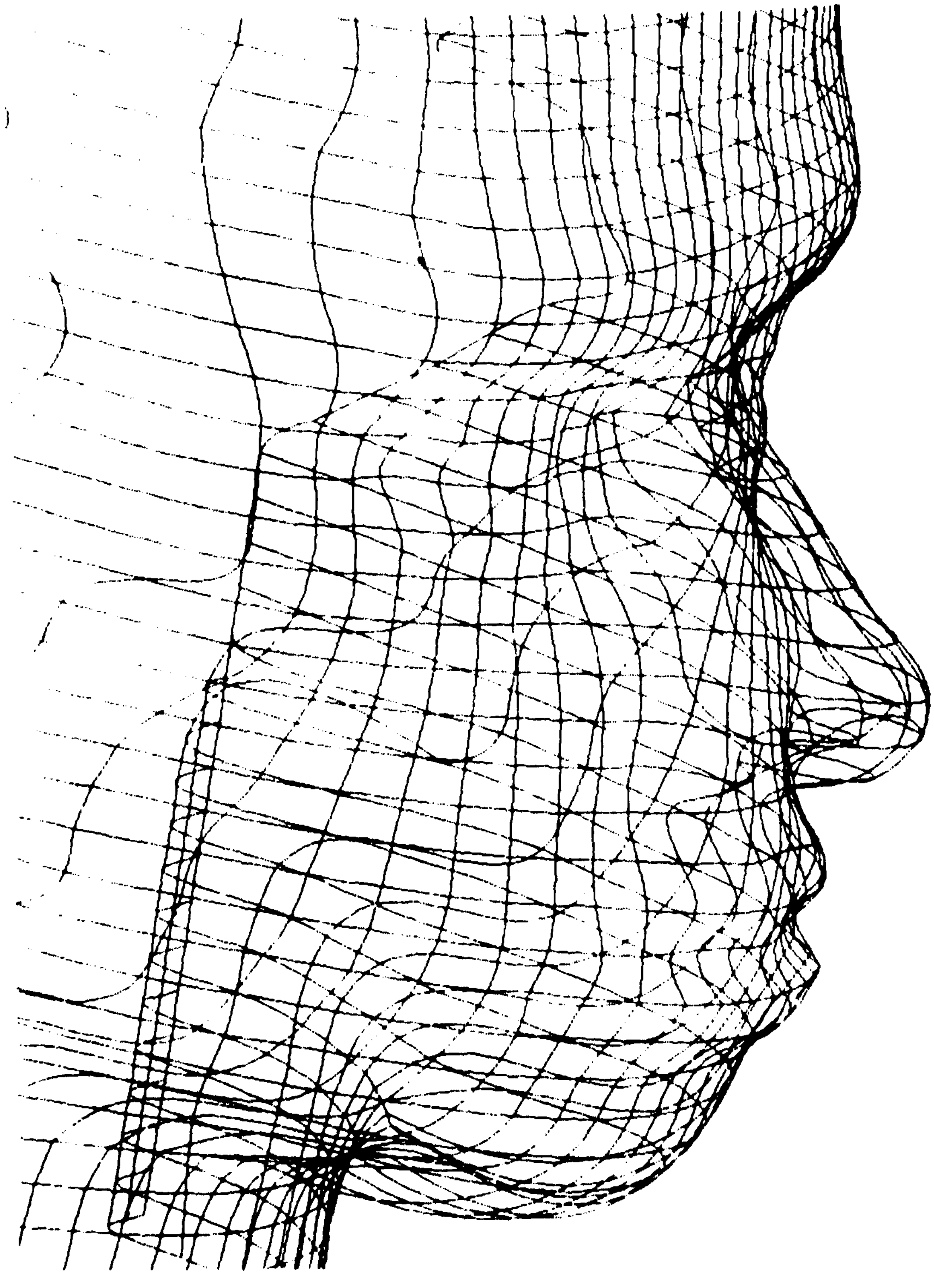


Figure 9: Spline-curve view of MULTI_PATCH.PRT (close-up)

APPENDIX A

```
$$   PROGRAM: NUSAMPL.GRS
$$
$$   OBJECT: TO EXTRACT EVERY N-TH POINT
$$   FROM A DATA FILE.  CREATE SCRATCH FILE
$$   TO RECEIVE SAMPLED DATA.
$$
$$    MVAL= NUMBER OF STEPS ON X-AXIS
$$    NVAL= NUMBER OF STEPS ON Y-AXIS
$$    TOTAL= NUMBER OF DATA POINTS IN SOURCE
$$
     STRING/FNAM(30)
     NUMBER/X(3),Y(3),Z(3),MAT1(12),TOTAL
     NUMBER/MVAL,NVAL
     DATA/FNAM,'@DPFO:DAVID:HEAD1DEC.TXT'
     DATA/TOTAL,17000,MVAL,3,NVAL,3
     MAT1= MATRIX/SCALE,1
$$
$$   DATA POINTS ARE STORED IN FILE 'FNAM'
$$
     FETCH/TXT,1,FNAM,IFERR,ERR1:
     RESET/1
$$
$$   CREATE FILE TO STORE SORTED POINTS
$$
     CREATE/TXT,2,'NUSAMPL1.TXT',$
     IFERR,ERR4:
$$
$$   READ IN THE POINTS FROM THE FILE
$$
     K= 1
     M= 1
     ZNU= Z(1)
     DO/L1:,I,1,TOTAL
     DO/L12:,J,1,3
     DELIM/' '
     READ/1,IFEND,L2:,IFERR,ERR2:,$
X(J),Y(J),Z(J)
     IF/(Z(J)-ZNU)>0.5,JUMP/L10:
  L12:
     JUMP/L11:
  L10:
     ZNU= Z(J)
     M= M+1
     IF/M<=NVAL,JUMP/L20:
     M= 1
  L20:
     K= K+1
     IF/(X(J)-X(1))>0.1,JUMP/L1:
     K= 1
```

APPENDIX A (cont'd)

```
  [illegible]1:
    IF/K<>1,JUMP/L1:
    IF/M<>NVAL,JUMP/L1:
    APPEND/2
    WRITE/2,USING,'#@@@@@@@.@@@@',$
Z(J),X(J),Y(J)
  L1:
$$
$$  STORE THE FILE IN ARCHIVES
$$
  L2:
    FTERM/TXT,1,IFERR,ERR3:
    FILE/TXT,2,'NUSAMPL1.TXT'
$$
  TERM:
    HALT
$$
$$  ERROR MESSAGES
$$
  ERR1:MESSG/'ERR1:',' ERROR IN FETCH'
       JUMP/TERM:
  ERR2:MESSG/'ERR2:',' READ ERROR
       JUMP/TERM:
  ERR3:MESSG/'ERR3:',' ERROR IN FILE TERMINATION'
       JUMP/TERM:
  ERR4:MESSG/'ERR4:',' CREATION ERROR'
       JUMP/TERM:
```

APPENDIX B

```
$$ PROGRAM:  POINTGEN.GRS
$$
$$ OBJECT:   TO GENERATE POINTS
$$ FROM A DATA FILE IN 'FNAM'
$$ AND DISPLAY THEM
$$
$$ 1)  DECLARE THE VARIABLES:  POINTS MUST
$$     BE READ IN
$$
    ENTITY/PT
    STRING/FNAME(30)
    NUMBER/X,Y,Z,MAT1(12)
    DATA/FNAM, "@DPFO:DAVID:NUSAMPL.TXT'
    DATA/TOTAL,3000
    MAT1= MATRIX/SCALE,1
$$
$$ 2)  DATA POINTS ARE STORED IN FILE 'FNAM'
$$     CREATE THE PART IN THE FILE
$$     STRUCTURE
$$
    CREATE/PART,'POINTGEN',INCHES,IFERR,ERR5:
    FETCH/TXT,1,FNAM,IFERR,ERR1:
    RESET/1
    &WCSDRW= &YES
    &WCSLNK= &NO
    &CSMODE= &TEMP
    &CNMODE= &MODEL
    VIEW/7
$$
$$ 3)  READ IN THE DATA POINTS FROM THE FILE
$$
    &SYSCLR= &YELLOW
  BEGN:
    DO/L1:,I,1,TOTAL
    READ/1,USING,'#@@@@@@@.@@@@',IFEND,$
L10:,IFERR,ERR2:,X,Y,Z
$$
$$ 4)  CREATE UNIGRAPHICS POINTS FROM
$$     THE DATA POINTS READ
$$
    PT=POINT/X,Y,Z
    &ENTCLR= &WHITE
  L1:
  L10:
    VIEWE/AUTO
    FTERM/TXT,1,IFERR,ERR3:
    FILE/PART,IFERR,ERR6:
$$
$$
```

APPENDIX B (cont'd)

```
  TERM:
    HALT
$$
$$ 5)  ERROR MESSAGES
$$
  ERR1:MESSG/'ERR1:',' ERROR IN FETCH'
        JUMP/TERM:
  ERR2:MESSG/'ERR2:',' READ ERROR'
        JUMP/TERM:
  ERR3:MESSG/'ERR3:',' TERM: ERROR'
        JUMP/TERM:
  ERR4:MESSG/'ERR4:',' FOUND END-OF-FILE'
        JUMP/TERM:
  ERR5:MESSG/'ERR5:',' CREATION ERROR'
        JUMP/TERM:
  ERR6:MESSG/'ERR6:',' FILING ERROR'
        JUMP/TERM:
```

APPENDIX C

```
$$  PROGRAM: MULTI_PATCH.GRS
$$
$$  THE PROGRAM READS A DATA FILE
$$  CREATES POINTS, SPLINE CURVES,
$$  AND AUTOMATES THE GENERATION OF
$$  MULTIPLE-PATCH SCULPTURED SURFACES
$$
$$  M= NUMBER OF POINTS ALONG X-AXIS
$$  N= NUMBER OF POINTS ALONG Y-AXIS
$$  U= NUMBER OF POINTS PER PRIMARY
$$     SPLINE -1
$$  W= NUMBER OF POINTS PER SECONDARY
$$     SPLINE -1
$$  RR= NUMBER OF PRIMARY CURVES PER
$$      SURFACE PATCH
$$  SS= NUMBER OF SECONDARY CURVES PER
$$      SURFACE PATCH
$$
$$  PTA(I,J)= TWO-DIMENSIONAL POINT ARRAY
$$  PATCH(P,Q)= POINT ARRAY SUBSET FOR
$$  EACH PATCH SURFACE
$$
     ENTITY/PT(1408),PTA(44,32),PCURV(44)$
,SCURV(32),SURF,GRPSLN(125),PATCH(7,7)
     STRING/FNAM(30),GNAM(30)
     NUMBER/X(1408),Y(1408),Z(1408),M,N,MAT1(12)$
,P,Q,V,W,RR,SS,TOTAL
     DATA/FNAM,'@DPFO:DAVID:NUSAMPL.TXT'
     DATA/M,44,N,32,RR,6,SS,4,V,7,W,7
     DATA/TOTAL,1408
     MAT1=MATRIX/SCALE,1
$$
$$  THE DATA POINTS ARE STORED IN THE
$$  FILE QUOTED IN 'FNAM'
$$  CREATE THE PART AND FETCH THE DATA
$$  FILE:
$$
     CREATE/PART,'MULTI_PATCH',INCHES,IFERR,ERR1:
$$
     FETCH/TXT,1,FNAM,IFERR,ERR2:
     RESET/1
$$
$$   SET THE DISPLAY FUNCTIONS:
$$
     &WCSDRW= &YES
     &WCSLNK= &NO
     &CSMODE= &TEMP
     &CNMODE= &MODEL
     VIEW/7
```

APPENDIX C (cont'd)

```
$$
$$   READ POINTS FROM SORTED DATA
$$   FILE
$$
     &SYSCLR= &YELLOW
  BEGN:
     DO/L1:,I,1,TOTAL
     READ/1,USING,'#@@@@@@@.@@@@',IFEND,L1:,IFERR,ERR3:,$
X(I),Y(I),Z(I)
$$
     &ENTCLR= &WHITE
     PT(I)= POINT/X(I),Y(I),Z(I)
  L1:
     VIEW/AUTO
$$
$$  CONSTRUCT A TWO-DIMENSIONAL POINT ARRAY
$$
     RESET/1
     K= 1
     DO/L3:,I,1,M
     DO/L2:,J,1,N
     READ/1,USING,'#@@@@@@@.@@@@',IFEND,L1:,4
IFERR,ERR3:,X(K),Y(K),Z(K)
     PTA(I,J)= POINT/X(K),Y(K),Z(K)
     K= K+1
  L2:
  L3:
$$
$$  SORT POINTS INTO SUB-SETS OF
$$  THE MAIN DATA ARRAY
$$
     S= 0
  L14:
     R= 0
  L13:
     P=1
$$
     DO/L11:,I,1,V
     K= I+R*V-R
$$
     Q= 1
     DO/L12:,J,1,W
     L= J+S*W-S
$$
     PATCH(P,Q)= PTA(K,L)
     Q= Q+1
  L12:
     P= P+1
```

APPENDIX C (cont'd)

```
  L11:
$$
$$  CONSTRUCT PRIMARY SPLINE CURVES
$$
      &ENTCLR= &RED
      DO/L4:,K,1,V
      PCURV(K)= SPLINE/PATCH(K,1..W)
  L4:
$$
$$  CONSTRUCT SECONDARY SPLINE CURVES
$$
      DO/L5:,KK,1,W
      SCURV(KK)= SPLINE/PATCH(1..V,KK)
  L5:
      &ENTCLR= &BLUE
      &SGRIDU= 1
      &SGRIDV= 1
$$
$$  CONSTRUCT SCULPTURED SURFACE
$$
      SURF= SSURF/PRIMA,PCURV(1..V),$
CROSS,SCURV(1..W)
$$
$$  TRANSFORM SPLINES
$$
      TRANSF/MAT1,PCURV(1..V),SCURV(1..$
W),MOVE
      R= R+1
      IF/R<=RR,JUMP/L13:
      S= S+1
      IF/S<=SS,JUMP/L14:
$$
  L10:
$$
$$  TERMINATE AND FILE PART
$$
      FTERM/TXT,1,IFERR,ERR4:
      FILE/PART,IFERR,ERR5:
$$
  L6:
  TERM:
    HALT
$$
$$  ERROR MESSAGES
$$
    ERR1:MESSG/'ERR1:','ERROR IN CREATION'
    ERR2:MESSG/'ERR2:','ERROR IN FETCH'
    ERR3:MESSG/'ERR3:','CANNOT READ DATA FILE PROPERLY'
    ERR4:MESSG/'ERR4:','FILE TERMINATION FROBLEM'
    ERR5:MESSG/'ERR5:','ERROR IN FILING PART'
```

DOCUMENT CONTROL DATA

(Security classification of title, body of abstract and indexing annotation must be entered when the overall document is classified)

1. ORIGINATOR (the name and address of the organization preparing the document. Organizations for whom the document was prepared, e.g. Establishment sponsoring a contractor's report, or tasking agency, are entered in section 8.)
Defence Research Establishment Ottawa
National Defence
Ottawa, Ontario K1A 0Z4

2. SECURITY CLASSIFICATION (overall security classification of the document, including special warning terms if applicable)
UNCLASSIFIED

3. TITLE (the complete document title as indicated on the title page. Its classification should be indicated by the appropriate abbreviation (S,C or U) in parentheses after the title.)
AN ALGORITHM FOR AUTOMATED SURFACE MODELLING IN UNIGRAPHICS (U)

4. AUTHORS (Last name, first name, middle initial)
HIDSON, David

5. DATE OF PUBLICATION (month and year of publication of document)
MARCH 1990

6a. NO. OF PAGES (total containing information. Include Annexes, Appendices, etc.)
29

6b. NO. OF REFS (total cited in document)
3

7. DESCRIPTIVE NOTES (the category of the document, e.g. technical report, technical note or memorandum. If appropriate, enter the type of report, e.g. interim, progress, summary, annual or final. Give the inclusive dates when a specific reporting period is covered.)
DREO Technical Note

8. SPONSORING ACTIVITY (the name of the department project office or laboratory sponsoring the research and development. Include the address.)
Defence Research Establishment Ottawa
National Defence
Ottawa, Ontario K1A 0Z4

9a. PROJECT OR GRANT NO. (if appropriate, the applicable research and development project or grant number under which the document was written. Please specify whether project or grant)
051LD

9b. CONTRACT NO. (if appropriate, the applicable number under which the document was written)

10a. ORIGINATOR'S DOCUMENT NUMBER (the official document number by which the document is identified by the originating activity. This number must be unique to this document.)
DREO TECHNICAL NOTE NO. 90-6

10b. OTHER DOCUMENT NOS. (Any other numbers which may be assigned this document either by the originator or by the sponsor)

11. DOCUMENT AVAILABILITY (any limitations on further dissemination of the document, other than those imposed by security classification)

(X) Unlimited distribution
( ) Distribution limited to defence departments and defence contractors; further distribution only as approved
( ) Distribution limited to defence departments and Canadian defence contractors; further distribution only as approved
( ) Distribution limited to government departments and agencies; further distribution only as approved
( ) Distribution limited to defence departments; further distribution only as approved
( ) Other (please specify):

12. DOCUMENT ANNOUNCEMENT (any limitation to the bibliographic announcement of this document. This will normally correspond to the Document Availabilty (11). However, where further distribution (beyond the audience specified in 11) is possible, a wider announcement audience may be selected.)

13. ABSTRACT (a brief and factual summary of the document. It may also appear elsewhere in the body of the document itself. It is highly desirable that the abstract of classified documents be unclassified. Each paragraph of the abstract shall begin with an indication of the security classification of the information in the paragraph (unless the document itself is unclassified) represented as (S), (C), or (U). It is not necessary to include here abstracts in both offical languages unless the text is bilingual).

Remote laser scanning of complex 3-D geometry can generate data sets with large numbers of points. This paper describes how these data are read into a CAD/CAM database (Unigraphics). A program is written in Graphics Interactive Programming Language (a subset of Unigraphics) that edits the data files, generates points, spline curve and surface entities in Unigraphics and stores the created models and parts in the databases. Programs for data sampling and editing are also shown separately.

14. KEYWORDS, DESCRIPTORS or IDENTIFIERS (technically meaningful terms or short phrases that characterize a document and could be helpful in cataloguing the document. They should be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location may also be included. If possible keywords should be selected from a published thesaurus. e.g. Thesaurus of Engineering and Scientific Terms (TEST) and that thesaurus-identified. If it is not possible to select indexing terms which are Unclassified, the classification of each should be indicated as with the title.)

COMPUTER-AIDED DESIGN
SURFACE MODELLING
PROGRAMMING
ANTHROPOMETRY
COMPLEX GEOMETRY